# The Manifesto Corpus: a new resource for research on political parties and quantitative text analysis

Merz, Nicolas; Regel, Sven; Lewandowski, Jirka

Merz, Nicolas; Regel, Sven; Lewandowski, Jirka



# The Manifesto Corpus: A new resource for research on political parties and quantitative text analysis

Research & Politics

*Research Article*

Research and Politics
April-June 2016: 1–8

DOI: 10.1177/2053168016643346
rap.sagepub.com
SAGE

# The Manifesto Corpus: A new resource for research on political parties and quantitative text analysis

**Nicolas Merz, Sven Regel and Jirka Lewandowski**

**Abstract**

This article presents a digital, open-access, multilingual, annotated corpus of electoral programs. It complements the recent methodological innovations in (semi-) computerized content analysis by providing a large, standardized text corpus for the political science community. The corpus is based on the collection of the Manifesto Project, which comprises of (at the time of writing) the largest hand-annotated text corpus of electoral programs available. Since 2009 the project's costly and time-intensive procedure of collecting and coding documents has been fully digitized. As a result, it now provides more than 1800 machine readable documents from 40 different countries. Six hundred of these documents contain content-analyzed annotations at the level of single (quasi-) sentences, which correspond to the Manifesto Project coding scheme. Additionally, the corpus will continually be extended by incorporating new elections and digitizing older documents. The database also provides meta-information for each document (eg. party, election, language, etc.) that allow it to be referenced back to the Manifesto Dataset. The corpus is stored in a standardized format in an online database, and an API and R package (*manifestoR*) guarantee easy access.

**Keywords**

Electoral programs, text corpus, R package

## Introduction

This article presents the Manifesto Corpus, a new text corpus consisting of digitized and coded electoral programs (Lehmann et al., 2016).[1] The corpus is based on the collection and coding of the Manifesto Project (Volkens et al., 2015). It is one of the largest human-annotated, open-access, cross-national text corpora in political science, and is the result of a long-term endeavor in digitizing and annotating electoral programs.

For decades the Manifesto Project (as the Manifesto Research Group from 1979 to 1989, the Comparative Manifestos Project (CMP) from 1989 to 2009 and as Manifesto Research on Political Representation (MARPOR) from 2009 onwards) has generated and distributed a data set based on the content analysis of electoral programs of the major political parties in (mainly) the OECD and Central and Eastern Europe. To generate the data set, trained native-language expert coders are asked to split the electoral programs into statements (so-called quasi-sentences) and to allocate to each statement some category of an extensive coding scheme of policy goals. Until now the published data set has provided the frequencies of these categories for each coded electoral program, and the most popular use of the data set has been to calculate the left–right positions of political parties from these data. It has now become one of the few data sets allowing the empirical test of theories of party competition both transnationally and over time, and since the first release of the data set it has been used in hundreds of studies on political parties, party systems, coalition building, agenda-setting and party strategies, among others.

Prior to 2009, the Manifesto data set only provided code frequencies at the document level, and information about

WZB Berlin Social Science Center, Berlin, Germany

**Corresponding author:**
Nicolas Merz, WZB Berlin Social Science Center, Reichpietschufer 50, 10785 Berlin Germany.
Email: nicolas.merz@wzb.eu

individual sentence codes within a document was, for the most part, inaccessible to users of the data set. In 2009 however, the infrastructure and coding processes of the project were digitized, which included the conversion of the documents to a machine-readable format and the implementation of a digitized document coding procedure. This data is now maintained and distributed as the Manifesto Corpus, comprising more than 1800 machine readable documents: among them more than 600 digitally coded documents and more than 600,000 annotated statements. The Manifesto Corpus is consistent with the trend of "treating text as data", by providing a large human-annotated standardized text corpus.

This article provides a description of the creation of the corpus, including the digitization, the annotation and the storage format. Access to the corpus is provided through a companion R package (*manifestoR*) and a web application programming interface (API). We illustrate the different uses of the Manifesto Corpus using several example cases and conclude with a discussion on drawbacks and future directions of research which apply and extend this new text corpus.

## The Manifesto Corpus

### *Collection, digitization and annotation*

The core of the Manifesto Corpus consists of the documents and codings of the Manifesto Data Collection (Budge et al., 2001; Klingemann et al., 2006; Volkens et al., 2015). The Manifesto Project collects and codes electoral programs for all relevant political parties at democratic elections from 1945 or the first democratic election in over 50 countries. A party is considered relevant if it is represented in parliament with at least one seat (in established democracies) or with two seats (in young democracies or countries with highly fragmented party systems). The country sample consists of most OECD countries and Central and Eastern European democracies. A full list of coverage can be found in Table 1. The collection of documents consists of electoral programs issued by parties before the election. In the few cases where parties do not publish electoral programs, proxy documents, such as party leader speeches, general party platforms, etc., are coded as substitute documents. Country experts (usually political scientists who are native language speakers) are hired to code the electoral programs. Coders first split the electoral programs into so-called "quasi-sentences", each of which "contains exactly one statement or message" (Werner et al., 2011). A natural sentence can contain multiple quasi-sentences, but a quasi-sentence can never span over more than one natural sentence. Natural sentences are split into quasi-sentences if they contain unrelated statements, possibly indicated by semi-colons, or if it is possible to allocate different codes to different parts of the natural sentence. The coders then allocate to every quasi-sentence a code, corresponding to one of 56 categories, which captures the most relevant policy issues and goals (for more information on the coding scheme, see the coding instructions and the dataset documentation in Werner et al. (2011) and Volkens et al. (2015). In order to do this, coders are taken through a training process, during which they receive extensive feedback from the coding supervisor. The training process has proven

**Table 1.** Coverage of the Manifesto Corpus: (1) machine-readable programs; (2) digitally coded programs; (3) digitally annotated quasi-sentences (Version 2016-1).

| Country | (1) | (2) | (3) |
|---|---|---|---|
| Armenia | 6 | 6 | 2038 |
| Australia | 78 | 19 | 8656 |
| Austria | 54 | 19 | 19913 |
| Belgium | 133 | 16 | 21679 |
| Bulgaria | 4 | 4 | 5872 |
| Canada | 48 | 5 | 4138 |
| Croatia | 30 | 30 | 12167 |
| Cyprus | 12 | 12 | 7050 |
| Czech Rep. | 21 | 21 | 17474 |
| Denmark | 175 | 36 | 7572 |
| Estonia | 13 | 13 | 5885 |
| Finland | 97 | 16 | 8165 |
| France | 53 | 10 | 4809 |
| Georgia | 11 | 11 | 3720 |
| Germany | 77 | 26 | 48523 |
| Great Britain | 39 | 3 | 2259 |
| Greece | 28 | 28 | 23956 |
| Hungary | 20 | 20 | 34446 |
| Iceland | 19 | 19 | 2277 |
| Ireland | 64 | 13 | 16290 |
| Italy | 95 | 14 | 4398 |
| Lithuania | 21 | 21 | 16979 |
| Luxembourg | 17 | 17 | 27955 |
| Macedonia | 30 | 30 | 40999 |
| Mexico | 11 | 11 | 9039 |
| Netherlands | 90 | 31 | 48408 |
| New Zealand | 75 | 27 | 7777 |
| Norway | 87 | 14 | 33559 |
| Poland | 11 | 11 | 11784 |
| Portugal | 58 | 9 | 10240 |
| Romania | 3 | 3 | 611 |
| Russia | 4 | 4 | 1506 |
| Serbia | 12 | 12 | 8081 |
| Slovakia | 21 | 21 | 13582 |
| Slovenia | 23 | 23 | 22982 |
| South Africa | 17 | 17 | 6423 |
| Spain | 71 | 37 | 61185 |
| Sweden | 95 | 15 | 7933 |
| Switzerland | 81 | 21 | 5437 |
| Turkey | 8 | 8 | 15706 |
| United States | 28 | 7 | 9236 |
| Total | 1840 | 680 | 620709 |

essential in ensuring a consistent understanding of the categories and coding scheme across countries and over time, and for an acceptable reliability of the coding process (Lacewell and Werner, 2013).

In the past the coding of these documents was performed using printed copies of the electoral programs, annotating in the margins of the pages. Coders simply summed the frequencies of the different codes and reported them to the coding supervisor who generated the data set from this information. The first serious effort towards digitization was made by Paul Pennings and Hans Keman of the Comparative *Electronic* Manifestos Project (2006), who digitized 1144 electoral programs included in the Manifesto Corpus. In 2009, the Manifesto project changed its infrastructure to incorporate a fully digital coding process, in which the procedures of splitting into quasi-sentences and coding are performed on the digitized text, allowing a link to be made between a code and a specific text segment. Currently, the Corpus contains 677 of these digitally coded documents.

### *Format, access and versions*

The Manifesto Corpus is currently stored in a digital data repository and can be browsed online or accessed with an open-source package for the statistical software R called *manifestoR* or via an API.

The corpus stores electoral documents in two main formats. The first is as a pdf document containing scans of the printed copies of the election programs or (in the case of more recent elections) the pdf files that have been downloaded from the parties' websites. Although the pdf documents are not machine-readable, they provide important information about the original layout of the document.

The second format is as a machine-readable document generated from the pdf documents by the Comparative Electronic Manifestos Project and by MARPOR. The texts are UTF-8 encoded to ensure that they are correct, compatible and accessible despite the wide range of different languages they contain (Lucas et al., 2015). Approximately one third of these machine-readable documents also contain additional information on unitizing and coding. The machine-readable documents described above differentiate the Manifesto Corpus from existing text archives, such as polidoc.net (Benoit et al., 2009). The following examples will mostly use these documents to illustrate why the digital alignment of code and text is so beneficial to future research.

In addition, every document in the corpus is linked to metadata about its language, document type, and the party and election it belongs to. This information also links the electoral programs to the Manifesto Project's main data set, which contains several additional variables concerning related information and data quality (eg. election results, coder reliability scores, etc.). The annotations of the Manifesto Corpus and the metadata allow for the filtering of the Corpus based on multiple criteria. As an example, it is possible to select only documents from a specific party, country or language, or only text segments related to specific issues or policy goals.

Access to the Manifesto Corpus is free. The corpus can be browsed using an online web application that provides functionality for the selection of specific programs, the filtering of text by codes/parties/election year, the downloading of the original pdfs or csv documents as well as full-text searches. Another way to access the Manifesto Corpus is using the software *manifestoR* (Lewandowski et al., 2015), an open source package for the free and open source statistical computing environment R. It provides routines for downloading single electoral programs, as well as for bulk downloading large subsets of the Manifesto Corpus according to user-defined criteria. The downloaded documents can be inspected manually or passed on to additional software for automated processing. To ensure seamless integration with popular software for natural language processing, text mining and data analysis, *manifestoR* uses the technical infrastructure provided by the popular R package *tm* (Feinerer and Hornik, 2015). manifestoR is available on the Comprehensive R Archive Network.[2] *manifestoR* is accompanied by detailed documentation.

The Manifesto API provides an even more general way to access the Manifesto Corpus. It can be queried for documents and metadata via HTTP and returns the requested information in JavaScript Object Notation (JSON), a standardized format also used by many other APIs. In this way, all information in the Manifesto Corpus can be accessed from almost any programming language the user may prefer. The API is documented and can be reached at the Manifesto Project's website (https://manifesto-project.wzb.eu).

The Manifesto Corpus is continually extended, updated and corrected. This, however, creates the problem that analyses conducted using previous versions of the corpus may not be able to be reproduced using later versions. To circumvent this, the Manifesto Corpus is stored using the versioning system *git*. This ensures that even minor changes to the corpus are transparent and preserve reproducibility. All users can easily access any version of the Manifesto Corpus ever published via manifestoR or the API. The versioning system also avoids the issue of users having to put large parts of the corpus in repositories, a procedure requested by journals in order to ensure the reproducibility of their research. Instead of doing this, they only have to indicate the corpus version used in their analyses in their script files.

## Applications

### *Term frequencies by language, issue, or party*

One of the simplest applications of the Manifesto Corpus is the calculation of word (or term) frequencies indicating


**Figure 1.** Most frequent terms within electoral programs 2012 from four Dutch parties on the issue of multiculturalism (codes 607 and 608).

how often certain words, combinations of words, or word stems appear in a text. Term-document matrices can be generated which indicate the frequency of certain terms within each document in a corpus. The calculation of term frequencies is an intuitive way to summarize large text corpora.

Figure 1 is an example of this. The figure is based on the electoral programs for the general elections of 2012 issued by the four most popular Dutch parties. The corpus underlying the figure was cleaned by automatically stripping numbers and punctuations, and was filtered to only contain words from quasi-sentences coded with the categories 607 (multiculturalism positive) and 608 (multiculturalism negative). The figure indicates the parties' different framing of the issue of multiculturalism. To give an example: one can clearly see that Geert Wilders' right-wing populist party's (PVV) very critical stance on multiculturalism associates these issues with Islam and its presumed incompatability with Dutch society (islam, islamitische, ideologie, totalitaire). In contrast, the social-democratic Labor Party (PvdA), with a position much more in favor of multiculturalism, frames the issue more in terms of education, culture and socialization (school, kinderen, culturele).

With a slightly more advanced setup, the Manifesto Corpus can be used to generate issue-specific lists of keywords (dictionaries) in multiple languages. Having dictionaries for the same issue in multiple languages means it is no longer necessary to translate all source texts into a common language before analysis, as had to be done in the past (Lucas et al., 2015; Pennings, 2011). The process of translating into a common language is either very time consuming or has to rely on automatic translation, the quality of which for some languages is still poor.

As a consequence of the CMP annotations being independent of the language of the document and the metadata of each document indicating the language it is written in, it is possible to filter a corpus consisting of documents written in multiple languages by the same set of specific issues and policy goals, extract just the text segments related to those issues or goals (in whatever language they may be written in) and generate a list of keywords for each language in the corpus. Thus, it is easy to create a set of multilingual dictionaries on a specific issue.

Table 2 indicates the results of such an exercise. The word stems are the result of a term frequency matrix derived from a corpus that contains only statements related to the environment (CMP code 501: environmental protection) in twelve different languages.[3] Moreover, we automatically deleted words that are popular in all other categories as well.

## *Text reuse: tracing policy ideas in electoral programs*

In the following example, we illustrate how text reuse methods can be applied to the Manifesto Corpus to study the drafting process of electoral programs. Text reuse refers to the issue concerning "how content from a single or multiple number of known sources can be reused either verbatim (word-for-word copy) or otherwise rewritten depending upon factors influencing the creation of a new document" (Clough, 2001). The availability of digital text has led to an increasing interest in text reuse. Plagiarism software, which detects whether authors have copied text passages from the work of others or themselves without citation, is a prominent example of attempting to address a text reuse problem. However text reuse approaches have also been used to address other substantive issues, such as whether and to what degree journalists use articles from press agencies (Gaizauskas et al., 2001) or press releases from parties and candidates (Grimmer, 2010; Meyer et al., 2015).Wilkerson et al. (2015) applied text reuse methods in the field of legislative studies to analyze which policy ideas that were proposed in thousands of different bills made it into law.

In this example we compare published, enacted versions of electoral programs to earlier draft versions of the same program to study how rank-and-file members at party conventions influence electoral programs. Such drafts are usually written by a specific committee or by the party leadership. The draft is then presented and discussed at party conventions where the rank-and-file members can propose changes to the program which (if not adopted by the party leadership), are put to vote. We focus here on the German SPD and Green parties' programs from 2013. Both organized party conventions where rank-and-file members could make amendments, although the degree of participation varies between the two parties (Hornsteiner, 2015). This approach could easily be applied to other cases where draft versions of electoral programs are available.

We analyze which coded quasi-sentences in the official, enacted (and digitally coded) program were already in the draft proposed by the leadership, and which statements were added at the party convention. By comparing the enacted

**Table 2.** Most unique word stems by language and issue domain (category 501: environmental protection).

| | environment |
|---|---|
| Danish | miljø, natur, landbrug, grøn, energi, økologisk, vedvar, fødevar, forbrug, omstilling |
| Dutch | dier, natur, verbod, milieu, welzijn, dierproev, biodiversiteit, landschap, natuurgebied, water |
| English | water, environment, environ, climat, wast, natur, emiss, conserv, pollut, green |
| Finnish | itämer, ympäristö, päästäoj, luono, ilmastonmuutoks, jät, uusiutuv, vähent, mets, maataloud |
| French | climat, lenviron, énerget, écolog, renouvel, énerg, dénerg, consomm, environnemental, naturel |
| German | energi, erneuerbar, umwelt, natur, tier, klimaschutz, nutzung, umsetz, energieeffizienz, landwirtschaft |
| Hungarian | környezet, környezetvédelm, környez, megújuló, természet, energiaforrás, természetes, energ, környezetvédel, állapot |
| Italian | energ, animal, produzion, energet, elettr, rinnov, rif, ambiental, are, incentiv |
| Norwegian | utslipp, natur, miljøvenn, vern, energi, miljø, bærekraft, avfall, biologisk, forbruk |
| Portuguese | ambient, energ, energét, natur, resídu, ambiental, águ, orden, sol, verd |
| Spanish | agu, ambiental, ambient, natural, energ, contamin, energet, uso, ecolog, residu |
| Swedish | utsläpp, östersjön, grön, miljö, hållb, energi, fossil, skydd, natur, förnyb |

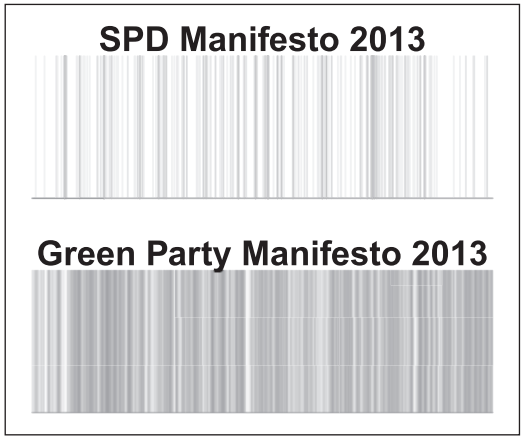


**Figure 2.** Text passages in the official electoral program that were added at party conventions (grey lines) compared to text passages already in the draft version (white).

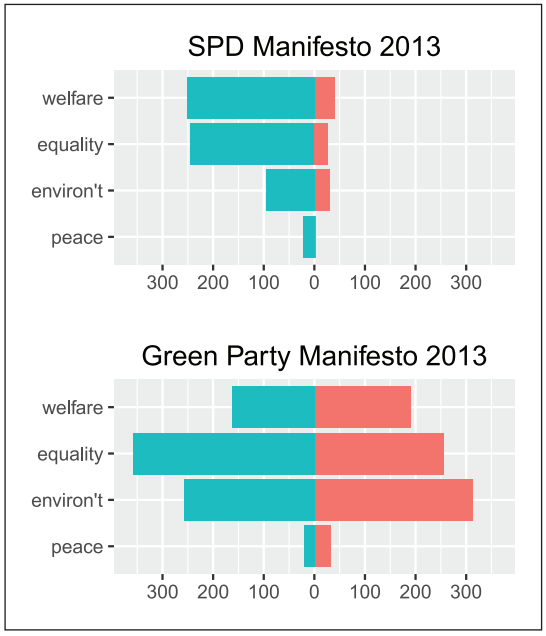


**Figure 3.** Absolute number of issue-specific statements in parties' electoral programs that were already in the draft version by the party leadership (turquoise) or added at the party convention by party rank-and-file members (red).

version with the draft, we can determine to what degree and in which issue areas the party convention changed the program. In order to do this, we check which quasi-sentences coded in the official program were already in the draft using an approximate string matching algorithm.

Figure 2 depicts the overall change from the draft version to the enacted program. A white line indicates a statement that was already in the draft version, a gray line indicates a statement that was added at the party convention. We can see that the amount of change differs drastically between parties. Where almost half of the statements in the Green's manifesto were added at the party convention, this is not the case for the SPD manifesto, where only 15% of all statements were added at the party convention.

Figure 3 shows the results of the comparison. Statements already included in the draft are shown on the left side in turquoise, and statements added at the party convention are shown on the right in red. We illustrate this here with four prominent issues, among them core issues of the SPD (welfare and equality) and the Green party (environmental protection and peace). The biggest differences can be clearly found between the parties, and not between issues. We cannot see a clear pattern indicating that party conventions cause change in core issues more than other issues.

## *Machine learning: training an automatic coding algorithm*

In the final example of an application we use the Manifesto Corpus for a semi-automatic coding task. For a long time, scholars have attempted to automate the coding of electoral programs and other political texts to avoid the high costs

associated with human coding. Scaling methods such as Wordscores (Laver et al., 2003) or Wordfish (Slapin and Proksch, 2008) scale documents or parts of documents along a latent dimension. The left–right scores derived from electoral programs by these methods correlate with measures of left–right scores derived from human coding of electoral programs. However, significant incongruences between both measures remain (Bräuninger et al., 2013). The advantage of Wordscores and Wordfish compared to human coding are the low costs of producing these scores. But these (almost) fully automatic scaling methods require intensive validity checks (Grimmer and Stewart, 2013) and can only provide party positions for overarching issues or dimensions such as left–right scores. In contrast to the scaling approaches, dictionary-based approaches were developed to make predictions about a large number of different categories. Pennings (2011) made a recent attempt with a new dictionary-based approach to score words in documents. His dictionary is originally based on the coding scheme of the Manifesto Coding Instructions. However, the creation of dictionaries is a very time-consuming task. Moreover, his approach is language specific and depends on the quality of automatic translations such as Google Translate. Instead, we suggest a semi-automatic coding approach that uses machine learning algorithms and human-annotated training data to annotate data *automagically*. Email spam filters are one of the most popular and widely used applications of such a classification task. In such a case, an algorithm decides whether incoming messages are spam or not based on messages that have been marked as spam in the past. Such classification tasks have also been used in political science for the classification of bills into a set of issues (Hillard et al., 2008). An advantage of this approach is that once the classifier is sufficiently trained, the costs of classification are almost zero. Moreover, a classifier can differentiate between several issues and therefore produces more fine-grained results than positions on latent dimensions, as in the case of Wordscores.

In this application, we use seventeen annotated electoral programs from the five major German parties from the elections 1998, 2002, 2005 and 2009 as a training set, containing a total of 27,942 quasi-sentences. We use the five electoral programs of the German federal election of 2013 as a test set. This simulates how accurate it would be to code future elections based on past annotations, a task that is quite relevant and is a plausible scenario. For our analysis we use RTextTools (Jurka et al., 2014), an R package for semi-automatic classification that facilitates the use and comparison of different machine-learning algorithms. We apply common pre-processing steps on the training set and on the test set, such as stop word removal, punctuation removal, and word stemming. As the quasi-sentences are sometimes very short and not understandable without context, we also used the three preceding and subsequent quasi-sentences of the focal quasi-sentence. We weighted the focal quasi-sentence by 1/2, the next closest sentences with 1/6, and the most distant with 1/12. We create a term-document matrix of unigrams, with a term-frequency inverse document-frequency weighting scheme (tfidf), a common practice in similar studies which puts more weight on rare, but distinct terms. We do not exclude any further (sparse) terms from the term-document matrix. We add a feature in the term-document matrix of the test and training set indicating the party issuing the programs as human coders are similarly aware of this. To classify the documents we use a Support Vector Machine (SVM, see Cortes and Vapnik, 1995). SVMs have been proven to be the most accurate and efficient algorithm for many different classification tasks, including tasks in political science (e.g. Hillard et al., 2008). To facilitate the comparison between computerized and human-coding we use the classifier to annotate the human-unitized quasi-sentences.

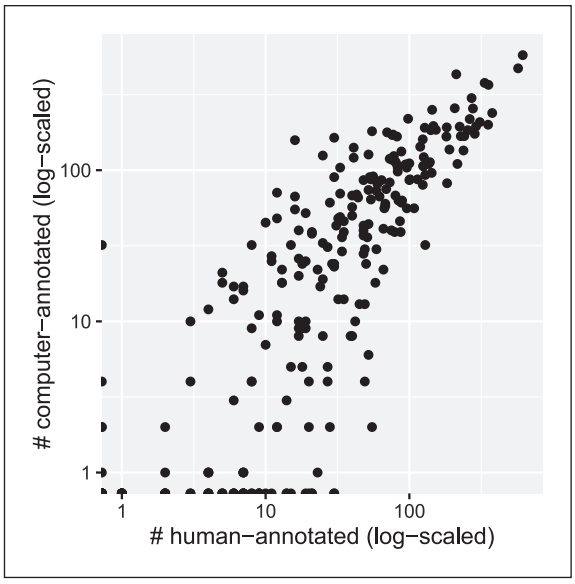


**Figure 4.** Comparison of code frequencies of 56 categories in five electoral programs by human and semi-automatic coding.

The classifier annotates 6779 of the 15,952 annotations (42%) with the same code as the human expert. At first, this seems disappointing. However, one has to take into account that the category scheme with its 56 categories is very complex and that also human coders produce agreement scores on the individual code level of only around 0.5, when compared with a master copy (for a discussion of human coder reliability in the Manifesto Project Dataset, see Lacewell and Werner, 2013; Mikhaylov et al., 2012). Moreover, some of this error cancels out when aggregating the codes on the quasi-sentence level to code frequencies. Figure 4 plots the frequency of categories in all five documents according to the human coder versus the classifier. The correlation of these scores is high and suggests a decent similarity of code assignment at the aggregate level (Pearson's r: 0.88, N=285).

The example we presented here is intended to be a proof of concept more than an in-depth classification study. The results presented are promising and illustrate the potentials of semi-automatic coding with the Manifesto Corpus as training data. However, there is a lot of room for

improvement and adaption to more fine-grained analyses (see Wiedemann, 2015).

### *Recoding subsets and adding code layers*

Finally, we would like to point out that researchers can now easily recode parts of the Manifesto Corpus. They can either recode the sentences of specific existing categories with a more fine-grained category scheme or add a new coding layer over all statements, combining them with the existing codes. We briefly illustrate both ways of recoding the Manifesto Corpus with two on-going projects that use the Manifesto Corpus.

Horn and van Kersbergen (2015) recode quasi-sentences from German electoral programs coded as 503 (Equality: Positive) and 504 (Welfare State: Expansion) with a more fine-grained sub-category scheme. These two categories are among the most frequently used. However, for scholars interested in the welfare state, these categories are too broad. Horn and van Kersbergen differentiate statements of equality into statements relating to the distribution of income and wealth, general statements on social equality, statements related to upward mobility in the sense of equal opportunities, and statements related to anti-discrimination/inclusion. They find that traditional left parties speak of equality more in terms of economic inequality, whereas right-wing parties tend to speak of equality in terms of anti-discrimination and inclusion.

Lehmann and Zobel (2015) conducted a pilot study demonstrating how adding a second layer of codes can provide insights into parties' framing strategies. They used crowdsourced coding (see also Benoit et al., forthcoming) to recode large parts of the Manifesto Corpus in regard to the issues of immigration and integration. As a consequence, they can analyze how parties connect the issue of immigration and integration of migrants with the policy goals coded using the existing coding scheme. They find that mainsteam parties tend to use more nationalist frames when talking about immigration and integration if a radical right party is represented in parliament.

## Discussion

As the Manifesto Corpus is based on the work of the Manifesto Project, it inherits some of the points of criticism related to the Manifesto Project's approach: such as the coding scheme (Zulianello, 2014), the use of proxy documents (Gemenis, 2012) and the reliability of the coding (Mikhaylov et al., 2012). The corpus cannot resolve all of these methodological problems. However, it greatly increases the transparency of the data production process, which may contribute something to the methodological debate on the validity and reliability of the Manifesto Dataset. Comparisons of human coding and automatic coding on the level of quasi-sentences could help to detect the weaknesses of both approaches similar to the comparisons of positions derived from expert ratings and electoral programs (Marks et al., 2007). Moreover, the Manifesto Corpus allows an in-depth and large scale study on the question of whether the coding of natural sentences is equally good compared to the coding of quasi-sentences as suggested by Däubler et al. (2012). But the Manifesto Corpus is first and foremost a new data source that allows substantive research questions to be answered that could not be answered before. It is a free digital text corpus based on the collection, digitization and coding of the Manifesto Project which offers multiple new and innovative ways to analyze electoral programs, only a few of which could be illustrated here.

### Acknowledgements

We are indebted to all our current and former colleagues in the MARPOR project, in particular to Pola Lehmann and Theres Matthieß whose contribution cannot be overstated. Moreover, we thank all the current and former student assistants and coders who did most of the heavy lifting in terms of digitization and coding. We presented an earlier version of this article in October 2014 at a workshop in Mannheim with the title "Political Context Matters: Content Analysis in the Social Sciences" and would like to thank all participants for fruitful comments and feedback. Finally, we want to acknowledge the work of the many contributors to the R statistical programming language and its ecosystem, without which neither the production nor the access to the Manifesto Corpus would be possible in the way it is now.

### Notes

1. When using the Manifesto Corpus please cite as Lehmann, Pola/Matthieß, Theres/Merz, Nicolas/Regel, Sven/Werner, Annika (2016) Manifesto Corpus. Version XXXX-X. WZB Berlin Social Science Center. Make sure to replace XXXX-X with the specific version of the Manifesto Corpus to ensure the replicability for your work. The corpus version used for all analyses presented here is 2016-1.
2. manifestoR can be downloaded at https://cran.r-project.org/web/packages/manifestoR/index.html or installed and loaded from within R by executing the following two commands: install.packages("manifestoR") library(manifestoR)
3. We selected here languages supported by the Snowball stemmer project. See http://snowball.tartarus.org/

### Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

### Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

### Carnegie Corporation of New York Grant

The open access article processing charge (APC) for this article was waived due to a grant awarded to Research & Politics from Carnegie Corporation of New York under its 'Bridging the Gap' initiative.

## References


Benoit K, Brauninger T and Debus M (2009) Challenges for estimating policy preferences: Announcing an open access archive of political documents. *German Politics* 18(3): 441–454.

Benoit K, Conway D, Lauderdale B, Laver M, et al. (forthcoming) Crowd-sourced text analysis: reproducible and agile production of political data. *American Political Science Review*.

Bräuninger T, Debus M and Müller J (2013) Estimating policy positions of political actors across countries and time. *Arbeitspapiere - Mannheimer Zentrum für Europäische Sozialforschung* 153. Available at: http://www.mzes.uni-mannheim.de/publications/wp/wp-153.pdf.

Budge I, Klingemann HD, Volkens A, et al. (2001) *Mapping Policy Preferences Estimates for Parties, Electors, and Governments 1945–1998*. Oxford: Oxford University Press.

Clough P (2001) Measuring text reuse in a journalistic domain. In: *Proc. of the 4th CLUK Colloquium*, pp.53–63.

Cortes C and Vapnik V (1995) Support-vector networks. *Machine Learning* 20(3): 273–297.

Däubler T, Benoit K, Mikhaylov S, et al. (2012) Natural sentences as valid units for coded political texts. *British Journal of Political Science* 42(04): 937–951.

Feinerer I and Hornik K (2015) tm: Text Mining Package. Available at: https://cran.r-project.org/web/packages/tm/index.html.

Gaizauskas R, Foster J, Wilks Y, et al. (2001) The METER corpus: A corpus for analysing journalistic text reuse. In: *Proceedings of the corpus linguistics 2001 conference*, pp.214–223.

Gemenis K (2012) Proxy documents as a source of measurement error in the Comparative Manifestos Project. *Electoral Studies* 31(3): 594–604.

Grimmer J (2010) A Bayesian hierarchical topic model for political texts: Measuring expressed agendas in senate press releases. *Political Analysis* 18(1): 1–35. DOI:10.1093/pan/mpp034 .

Grimmer J and Stewart BM (2013) Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis* 21(3): 267–297.

Hillard D, Purpura S and Wilkerson J (2008) Computer-assisted topic classification for mixed-methods social science research. *Journal of Information Technology & Politics* 4(4): 31–46.

Horn A and van Kersbergen K (2015) Peeping at the Corpus. What is really going on behind the equality and welfare items of the Manifesto Project? In: *Manifesto Project user conference*, 4–5 June 2015. Berlin.

Hornsteiner M (2015) Party manifestos in representative democracy: Strengthening the electoral connection? In: D'Ottavio G and Saalfeld T (eds) *Germany After the 2013 Elections*. Ashgate, Farnham.

Jurka TP, Collingwood L, Boydstun AE, et al. (2014) RTextTools: Automatic text classification via supervised learning. Available at: https://cran.r-project.org/web/packages/RTextTools/index.html.

Klingemann HD, Volkens A, Bara J, et al. (2006) *Mapping Policy Preferences II: Estimates for Parties, Electors, and Governments in Central and Eastern Europe, European Union and OECD 1990–2003*, vol. 2. Oxford: Oxford University Press.

Lacewell OP and Werner A (2013) Coder training: Key to enhancing coding reliability and estimate validity. In: Volkens A, Bara J, Budge I, et al. (eds) *Mapping Policy Preferences from Texts. Statistical Solutions for Manifesto Analysts*. Oxford: Oxford University Press.

Laver M, Benoit K and Garry J (2003) Extracting policy positions from political texts using words as data. *American Political Science Review* 97(02): 311–331.

Lehmann P, Matthieß T, Merz N, et al. (2016) Manifesto Corpus. *Version* 2016-1. WZB Berlin Social Science Center.

Lehmann P and Zobel M (2015) A question of national pride or universal rights: How parties frame immigration issues. In *73th annual meeting of the Midwest Political Science Association*, 16–19 April 2015. Chicago, IL.

Lewandowski J, Merz N, Regel S, et al. (2015) manifestoR: Access and process data and documents of the Manifesto Project. Available at: https://cran.r-project.org/web/packages/manifestoR/index.html.

Lucas C, Nielsen RA, Roberts ME, et al. (2015) Computer-assisted text analysis for comparative politics. *Political Analysis* 23(2): 254–277. DOI: 10.1093/pan/mpu019.

Marks G, Hooghe L, Steenbergen MR, et al. (2007) Crossvalidating data on party positioning on European integration. *Electoral Studies* 26(1): 23–38.

Meyer T, Haselmayer M and Wagner M (2015) The media's gatekeeping function means that party press coverage often reproduces and reinforces existing power structures. Available at: http://www.democraticaudit.com/?p=15261.

Mikhaylov S, Laver M and Benoit KR (2012) Coder Reliability and misclassification in the human coding of party manifestos. *Political Analysis* 20(1): 78–91.

Pennings P (2011) Assessing the 'Gold Standard' of party policy placements: Is computerized replication possible? *Electoral Studies* 30(3): 561–570.

Pennings P and Keman H (2006) Comparative Electronic Manifestos Project. In cooperation with the Social Science Research Centre Berlin (Andrea Volkens, Hans-Dieter Klingemann), the Zentralarchiv für empirische Sozialforschung (GESIS), and the Manifesto Research Group (chairman: Ian Budge).

Slapin JB and Proksch SO (2008) A scaling model for estimating time-series party positions from texts. *American Journal of Political Science* 52(3): 705–722.

Volkens A, Lehmann P, Matthieß T, et al. (2015) *The Manifesto Data Collection. Manifesto Project (MRG/CMP/MARPOR). Version 2015a*. Wissenschaftszentrum Berlin für Sozialforschung (WZB).

Werner A, Lacewell O and Volkens A (2011) Manifesto coding instructions. 4th fully revised edition. Available at: https://manifestoproject.wzb.eu/down/papers/handbook_2011_version_4.pdf.

Wiedemann G (2015) Automatic classification for content analysis on german cmp data sets. In: *Manifesto Project user conference*, 4–5 June 2015. Berlin, Germany.

Wilkerson J, Smith D and Stramp N (2015) Tracing the flow of policy ideas in legislatures: A text reuse approach. *American Journal of Political Science* 59(4): 943–956.

Zulianello M (2014) Analyzing party competition through the comparative manifesto data: Some theoretical and methodological considerations. *Quality & Quantity* 48(3): 1723–1737.